WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Sunday, April 29, 2007

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR	
□	L17	L16 and cach$4 same (hash$3 or key or ind$3)	335
□	L16	L15 and (optimiz$5 or profil$4)	632
□	L15	L12 and (hash$4 or key or index)	1420
□	L14	L13 and (hash$4 or key or index)	94
□	L13	L12 and binary with (translat$4) and (cach$3)	156
□	L12	(translat$4) same (subject or foreign or target or host or destinat$4 or designat$4) same (cach$3)	2459
□	L11	(translat$4) same (subject or foreign or target or host or destinat$4 or designat$4) same (cach$3) same (hash)	59
□	L10	manchester.asn. and (cach$3)	5
□	L9	manchester.asn. and (cach$3) same (hash)	0
□	L8	transitive.asn.	46
□	L7	transitive.asn. and (cach$3)	6
□	L6	(hash) same (name or identifier or id) same (location or offset or length) same time$1 same (version or address)	216
□	L5	"710"/$.ccls. and (cach$3) and (hash)	296
□	L4	711/3,113,118-146,216.ccls. and (translat$4) and (cach$3) same (hash$3)	254
□	L3	717/136-161.ccls. and (translat$4) and (cach$3) same (hash$3)	38
□	L2	(translat$4) same (cach$3) same (hash)	203
□	L1	"20020059268" or "20010049818" or "6397242".pn. or "5768593".pn. or "20020133810" or "6615300".pn.	12

END OF SEARCH HISTORY

Sign in

Google

Web Images Video News Maps **more »**

binary translation cache hash | Search | Advanced Search
Preferences

Web

Results **1** - **10** of about **957,000** for **binary translation cache hash**. (**0.18** seconds)

[PDF] Method and apparatus for determining branch addresses in programs ...
File Format: PDF/Adobe Acrobat - View as HTML
reclaiming of translations from a dynamic **translation cache**. In static **binary translation**, it can be. used to detect previously undetected entry points into ...
www.research.ibm.com/vliw/Pdf/yo898334.pdf - Similar pages

[PDF] IBM Research Report
File Format: PDF/Adobe Acrobat - View as HTML
have solved them. We also present some preliminary. statistics on the effectiveness of **binary translation**. ... **cache** is implemented as a software **hash** table ...
www.research.ibm.com/.../RC23262%20-%20Full%20System%20**Binary**%20**Translation**%20-%20RISC%20to%20VLIW.pdf - Similar pages

Method and system for implementing subroutine calls and returns in ...
One way to ensure this is for the **binary translator** to store the OL instructions in a **translation cache** located in an area of memory isolated from, ...
www.freepatentsonline.com/6711672.html - 92k - Cached - Similar pages

[PDF] HDTrans: A Low-Overhead Dynamic **Translator**
File Format: PDF/Adobe Acrobat - View as HTML
been the use of dynamic **binary translation** to address numer- ... The return **cache** is a fixed. size D-space direct mapped **hash** table, indexed by a simple ...
www.eecs.harvard.edu/~prash/papers/wbia2005/wbia2005-paper.pdf - Similar pages

[PDF] A Low-Complexity Dynamic **Translator** for x86
File Format: PDF/Adobe Acrobat - View as HTML
Machine-level dynamic **binary translation** has been used in applications ranging from ...
Like Dynamo [8], VDebug uses a **hash**-based lookup scheme to **translate** ...
www.eecs.harvard.edu/~prash/papers/masters-thesis/masters-thesis-2.pdf - Similar pages

[PDF] Security Applications of Dynamic **Binary Translation**
File Format: PDF/Adobe Acrobat - View as HTML
transformers, and fragment **cache**. One of the goals in the development of SIND is the. evaluation of techniques for dynamic **binary translation**. ...
www.sandia.gov/iorta/docs/ddz_thesis.pdf - Similar pages

[PDF] Approaches for Universal Static **Binary Translation**
File Format: PDF/Adobe Acrobat - View as HTML
purpose if the **translator** does not **cache** the translated code, or the **cache** ... **hash** table, and one function that generates a **binary** operation will also be ...
www.cs.drexel.edu/static/reports/DU-CS-06-02.pdf - Similar pages

[PDF] Author Guidelines for 8
File Format: PDF/Adobe Acrobat
Dynamic **Binary Translation** and Optimization in a Whole-System Emulator -. - SkyEye ...
is to use **hash** table to replace list and use recent access ...
doi.ieeecomputersociety.org/10.1109/ICPPW.2006.32 - Similar pages

[PDF] Atome - **Binary Translation** for Accurate Simulation
File Format: PDF/Adobe Acrobat - View as HTML

cially designed for it, the Simulator **Translation Cache** (STC) [MW95]. It. will store translated memory addresses, using a **hash** table designed after the ...
www.cs.umu.se/~ens03mbs/report.pdf - Similar pages

[PDF] DIGITAL FX!32: Combining Emulation and BinaryTranslation
File Format: PDF/Adobe Acrobat - View as HTML
this combination of emulation and **binary translation**. In this paper, we describe how ... data by inserting values in a **hash** table whenever a rel- ...
research.compaq.com/wrl/DECarchives/DTJ/DTJP01/DTJP01PF.PDF - Similar pages

Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

binary translation cache hash | Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

acm PORTAL

USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

(binary and translat*) and cache and hash* SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used binary and translat and cache and hash

Found **9,284** of **200,192**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200

Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale

1 Managing bounded code caches in dynamic binary optimization systems

Kim Hazelwood, Michael D. Smith

September 2006 **ACM Transactions on Architecture and Code Optimization (TACO)**, Volume 3 Issue 3

Publisher: ACM Press

Full text available: pdf(666.72 KB) Additional Information: full citation, abstract, references, index terms

Dynamic binary optimizers store altered copies of original program instructions in software-managed code caches in order to maximize reuse of transformed code. Code caches store code blocks that may vary in size, reference other code blocks, and carry a high replacement overhead. These unique constraints reduce the effectiveness of conventional cache management policies. Our work directly addresses these unique constraints and presents several contributions to the code-cache management problem. ...

Keywords: Dynamic optimization, code caches, dynamic translation, just-in-time compilation

2 Hashing, indexing & parallelism: B-tree indexes, interpolation search, and skew

Goetz Graefe

June 2006 **Proceedings of the 2nd international workshop on Data management on new hardware DaMoN '06**

Publisher: ACM Press

Full text available: pdf(129.99 KB) Additional Information: full citation, abstract, references

Recent performance improvements in storage hardware have benefited bandwidth much more than latency. Among other implications, this trend favors large B-tree pages. Recent performance improvements in processor hardware also have benefited processing bandwidth much more than memory latency. Among other implications, this trend favors adding calculations if they save cache faults.With small calculations guiding the search directly to the desired key, interpolation search complements these trends m ...

3 Incremental computation via function caching

W. Pugh, T. Teitelbaum

January 1989 **Proceedings of the 16th ACM SIGPLAN-SIGACT symposium on Principles of programming languages POPL '89**

Publisher: ACM Press

Full text available: Additional Information:

pdf(1.55 MB) full citation, references, citings, index terms

4 Consistent hashing and random trees: distributed caching protocols for relieving hot spots on the World Wide Web

David Karger, Eric Lehman, Tom Leighton, Rina Panigrahy, Matthew Levine, Daniel Lewin

May 1997 **Proceedings of the twenty-ninth annual ACM symposium on Theory of computing STOC '97**

Publisher: ACM Press

Full text available: pdf(1.73 MB) Additional Information: full citation, references, citings, index terms

5 Verification techniques for cache coherence protocols

Fong Pong, Michel Dubois

March 1997 **ACM Computing Surveys (CSUR)**, Volume 29 Issue 1

Publisher: ACM Press

Full text available: pdf(1.25 MB) Additional Information: full citation, abstract, references, citings, index terms

In this article we present a comprehensive survey of various approaches for the verification of cache coherence protocols based on state enumeration, (symbolic model checking, and symbolic state models. Since these techniques search the state space of the protocol exhaustively, the amount of memory required to manipulate that state information and the verification time grow very fast with the number of processors and the complexity of the protocol mechanism ...

Keywords: cache coherence, finite state machine, protocol verification, shared-memory multiprocessors, state representation and expansion

6 Fast address lookups using controlled prefix expansion

V. Srinivasan, G. Varghese

February 1999 **ACM Transactions on Computer Systems (TOCS)**, Volume 17 Issue 1

Publisher: ACM Press

Full text available: pdf(258.60 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Internet (IP) address lookup is a major bottleneck in high-performance routers. IP address lookup is challenging because it requires a longest matching prefix lookup. It is compounded by increasing routing table sizes, increased traffic, higher-speed links, and the migration to 128-bit IPv6 addresses. We describe how IP lookups and updates can be made faster using a set of of transformation techniques. Our main technique, controlled prefix expansion, transf ...

Keywords: Internet address lookup, binary search on levels, controlled prefix expansion, expanded tries, longest-prefix match, multibit tries, router preformance

7 Scalable high-speed prefix matching

Marcel Waldvogel, George Varghese, Jon Turner, Bernhard Plattner

November 2001 **ACM Transactions on Computer Systems (TOCS)**, Volume 19 Issue 4

Publisher: ACM Press

Full text available: pdf(933.02 KB) Additional Information: full citation, abstract, references, citings, index terms

Finding the longest matching prefix from a database of keywords is an old problem with a number of applications, ranging from dictionary searches to advanced memory management to computational geometry. But perhaps today's most frequent best matching prefix lookups occur in the Internet, when forwarding packets from router to router. Internet traffic volume and link speeds are rapidly increasing; at the same time, a growing user population is increasing the size of routing tables against which p ...

Keywords: collision resolution, forwarding lookups, high-speed networking

8 Effect of node size on the performance of cache-conscious B$^+$-trees
Richard A. Hankins, Jignesh M. Patel
June 2003 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 2003 ACM SIGMETRICS international conference on Measurement and modeling of computer systems SIGMETRICS '03**, Volume 31 Issue 1
Publisher: ACM Press
Full text available: pdf(271.16 KB) Additional Information: full citation, abstract, references, index terms

In main-memory databases, the number of processor cache misses has a critical impact on the performance of the system. Cache-conscious indices are designed to improve performance by reducing the number of processor cache misses that are incurred during a search operation. Conventional wisdom suggests that the index's node size should be equal to the cache line size in order to minimize the number of cache misses and improve performance. As we show in this paper, this design choice ignores additi ...

Keywords: B$^+$-tree, cache-conscious, index

9 Implementing sorting in database systems
Goetz Graefe
September 2006 **ACM Computing Surveys (CSUR)**, Volume 38 Issue 3
Publisher: ACM Press
Full text available: pdf(518.63 KB) Additional Information: full citation, abstract, references, index terms

Most commercial database systems do (or should) exploit many sorting techniques that are publicly known, but not readily available in the research literature. These techniques improve both sort performance on modern computer systems and the ability to adapt gracefully to resource fluctuations in multiuser operations. This survey collects many of these techniques for easy reference by students, researchers, and product developers. It covers in-memory sorting, disk-based external sorting, and cons ...

Keywords: Key normalization, asynchronous read-ahead, compression, dynamic memory resource allocation, forecasting, graceful degradation, index operations, key conditioning, nested iteration

10 External memory algorithms and data structures: dealing with **massive data**
Jeffrey Scott Vitter
June 2001 **ACM Computing Surveys (CSUR)**, Volume 33 Issue 2
Publisher: ACM Press
Full text available: pdf(828.46 KB) Additional Information: full citation, abstract, references, citings, index terms

Data sets in large applications are often too massive to fit completely inside the computers internal memory. The resulting input/output communication (or I/O) between fast internal memory and slower external memory (such as disks) can be a major performance bottleneck. In this article we survey the state of the art in the design and analysis of external memory (or EM) algorithms and data structures, where the goal is to exploit locality in order to reduce the I/O costs. We consider a varie ...

Keywords: B-tree, I/O, batched, block, disk, dynamic, extendible hashing, external memory, hierarchical memory, multidimensional access methods, multilevel memory, online, out-of-core, secondary storage, sorting

11 Faster IP lookups using controlled prefix expansion
V. Srinivasan, George Varghese
June 1998 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1998 ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems SIGMETRICS '98/PERFORMANCE '98**, Volume 26 Issue 1
Publisher: ACM Press
Full text available: pdf(1.31 MB) Additional Information: full citation, abstract, references, citings, index terms

Internet (IP) address lookup is a major bottleneck in high performance routers. IP address lookup is challenging because it requires *a longest matching prefix* lookup. It is compounded by increasing routing table sizes, increased traffic, higher speed links, and the migration to 128 bit IPv6 addresses. We describe how IP lookups can be made faster using a new technique called *controlled prefix expansion*. Controlled prefix expansion, together with optimization techniques based on dyn ...

12 Configuration management & security: Secure sharing between untrusted users in a transparent source/binary deployment model
Eelco Dolstra
November 2005 **Proceedings of the 20th IEEE/ACM international Conference on Automated software engineering ASE '05**
Publisher: ACM Press
Full text available: pdf(276.98 KB) Additional Information: full citation, abstract, references, index terms

The Nix software deployment system is based on the paradigm of *transparent source/binary deployment*: distributors deploy descriptors that build components from source, while client machines can transparently optimise such source builds by downloading pre-built binaries from remote repositories. This model combines the simplicity and flexibility of source deployment with the efficiency of binary deployment. A desirable property is *sharing* of components: if multiple users install fro ...

Keywords: configuration management, hash rewriting, secure sharing, security, software deployment, source deployment

13 Scalable high speed IP routing lookups
Marcel Waldvogel, George Varghese, Jon Turner, Bernhard Plattner
October 1997 **ACM SIGCOMM Computer Communication Review , Proceedings of the ACM SIGCOMM '97 conference on Applications, technologies, architectures, and protocols for computer communication SIGCOMM '97**, Volume 27 Issue 4
Publisher: ACM Press
Full text available: pdf(1.66 MB) Additional Information: full citation, abstract, references, citings, index terms

Internet address lookup is a challenging problem because of increasing routing table sizes, increased traffic, higher speed links, and the migration to 128 bit IPv6 addresses. IP routing lookup requires computing the best matching prefix, for which standard solutions like hashing were believed to be inapplicable. The best existing solution we know of, BSD radix tries, scales badly as IP moves to 128 bit addresses. Our paper describes a new algorithm for best matching prefix using binary search o ...

14 A case for two-way skewed-associative caches
André Seznec
May 1993 **ACM SIGARCH Computer Architecture News , Proceedings of the 20th annual international symposium on Computer architecture ISCA '93**, Volume 21 Issue 2
Publisher: ACM Press
Full text available: pdf(975.20 KB) Additional Information: full citation, references, citings, index terms

15 Hardware Support for Control Transfers in Code Caches
Ho-Seop Kim, James E. Smith
December 2003 **Proceedings of the 36th annual IEEE/ACM International Symposium on Microarchitecture MICRO 36**
Publisher: IEEE Computer Society
Full text available: pdf(315.74 KB) Additional Information: full citation, abstract, citings, index terms

Many dynamic optimization and/or binary translationsystems hold optimized/translated superblocks in a codecache. Conventional code caching systems suffer fromoverheads when control is transferred from one cachedsuperblock to another, especially via register-indirectjumps. The basic problem is that instruction addresses inthe code cache are different from those in the original programbinary. Therefore, performance for register-indirectjumps depends on the ability to translate efficiently fromsour ...

16 Query evaluation techniques for large databases
Goetz Graefe
June 1993 **ACM Computing Surveys (CSUR)**, Volume 25 Issue 2
Publisher: ACM Press
Full text available: pdf(9.37 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Database management systems will continue to manage large data volumes. Thus, efficient algorithms for accessing and manipulating large sets and sequences will be required to provide acceptable performance. The advent of object-oriented and extensible database systems will not solve this problem. On the contrary, modern data models exacerbate the problem: In order to manipulate large sets of complex objects as efficiently as today's database systems manipulate simple records, query-processi ...

Keywords: complex query evaluation plans, dynamic query evaluation plans, extensible database systems, iterators, object-oriented database systems, operator model of parallelization, parallel algorithms, relational database systems, set-matching algorithms, sort-hash duality

17 CDNs and caching: Scalable techniques for memory-efficient CDN simulations
Purushottam Kulkarni, Prashant Shenoy, Weibo Gong
May 2003 **Proceedings of the 12th international conference on World Wide Web WWW '03**
Publisher: ACM Press
Full text available: pdf(238.28 KB) Additional Information: full citation, abstract, references; index terms

Since CDN simulations are known to be highly memory-intensive, in this paper, we argue the need for reducing the memory requirements of such simulations. We propose a novel memory-efficient data structure that stores cache state for a small subset of popular objects accurately and uses approximations for storing the state for the remaining objects. Since popular objects receive a large fraction of the requests while less frequently accessed objects consume much of the memory space, this approach ...

Keywords: approximate data structures, content distribution networks, simulation, web proxy cache

18 Routing with a clue
Anat Bremler-Barr, Yehuda Afek, Sariel Har-Peled
August 1999 **ACM SIGCOMM Computer Communication Review , Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication SIGCOMM '99**, Volume 29 Issue 4
Publisher: ACM Press
Full text available: pdf(1.26 MB) Additional Information: full citation, abstract, references, citings, index terms

We suggest a new simple forwarding technique to speed-up IP destination address lookup. The technique is a natural extension of IP, requires 5 bits in the IP header (IPv4, 7 in IPv6) and performs IP lookup nearly as fast as IP/Tag-switching but with a smaller memory requirement and a much simpler protocol. The basic idea is that each router adds a "clue" to each packet, telling its downstream router where it ended the IP lookup. Since the forwarding tables of neighboring routers are similar, the ...

19 Query processing: Cache-oblivious string B-trees
Michael A. Bender, Martin Farach-Colton, Bradley C. Kuszmaul
June 2006 **Proceedings of the twenty-fifth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems PODS '06**
Publisher: ACM Press
Full text available: pdf(154.30 KB) Additional Information: full citation, abstract, references, citings, index terms

B-trees are the data structure of choice for maintaining searchable data on disk. However, B-trees perform suboptimally

- when keys are long or of variable length,
- when keys are compressed, even when using *front compression*, the standard B-tree compression scheme,
- for range queries, and
- with respect to memory effects such as disk prefetching.

This paper presents a *cache-oblivious string B-tree* (COSB-tree) data structure that is efficient in al ...

Keywords: cache oblivious string B-tree, locality preserving front compression, packed-memory array, range query, rebalance

20 The design of a cache-friendly BDD library
David E. Long
November 1998 **Proceedings of the 1998 IEEE/ACM international conference on Computer-aided design ICCAD '98**
Publisher: ACM Press
Full text available: pdf(660.00 KB) Additional Information: full citation, references, citings, index terms

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Sign in

Google

Web Images Video News Maps **more »**

translation cache hash [Search] Advanced Search
Preferences

Web

Results **11** - **20** of about **1,150,000** for translation cache hash. (**0.18** seconds)

[PDF] A Low-Complexity Dynamic **Translator** for x86
File Format: PDF/Adobe Acrobat - View as HTML
conditional branch already exists in the trace **cache**, VDebug emits a conditional ... Like Dynamo [8], VDebug uses a **hash**-based lookup scheme to **translate** ...
www.eecs.harvard.edu/~prash/papers/masters-thesis/masters-thesis-2.pdf - Similar pages

CPU **cache** - Wikipedia, the free encyclopedia
These hints are a subset or **hash** of the virtual tag, and are used for selecting the way of the **cache** from which to get data and a physical tag. ...
en.wikipedia.org/wiki/CPU_**cache** - 74k - Cached - Similar pages

[PS] Software Based Instruction Caching for the RAW Architecture
File Format: Adobe PostScript - View as Text
based instruction **cache**. First, the **translation cache** is typically stored in the main memory ... Since an instruction **cache hash** table will need to have ...
web.mit.edu/jasonm/www/thesis.ps - Similar pages

[PDF] PHONE NUMBER **TRANSLATION** DELAY IN PCS SYSTEMS WITH ATM BACKBONES
File Format: PDF/Adobe Acrobat
introduced by the NGPN **translation** scheme we have proposed. ... the mapping is found (a **cache** hit), a **hash** and a query to the TS is avoided; ...
www.springerlink.com/index/t357x81q52393475.pdf - Similar pages

Virtual Memory in the IA-64 Linux Kernel > **Translation** Lookaside ...
The third, the virtual **hash** page table walker (VHPT walker) is designed to ... In contrast, insertion into a **translation cache** requires specification of ...
www.phptr.com/articles/article.asp?p=29961&seqNum=4 - 61k - Cached - Similar pages

[PPT] Address **Translation**
File Format: Microsoft Powerpoint - View as HTML
Variable size **hash** table; Processor register points to **hash** table base and gives ... Use a **cache** to speed up access; **Translation** Lookaside Buffer (TLB) ...
www.cs.wisc.edu/~remzi/Classes/537/Fall2005/Lectures/lecture14.ppt - Similar pages

Hash Routing Architecture: Briefing on Super Proxy Script
National University of Singapore is running 4 way Super Proxy (directory **hash** version) for network with 35000 students and staffs. Total **cache** size amounts ...
naragw.sharp.co.jp/sps/sps-e.html - 32k - Cached - Similar pages

Paul Sandhu's Weblog
These structures are arranged in a series of **hash** buckets with a hashing function ... Also, it would be more **cache** friendly if large mappings were grouped, ...
blogs.sun.com/paulsan/feed/entries/rss - 16k - Cached - Similar pages

Lecture notes for CSC 252, Thurs. Apr. 12, 2007ff Announcements A6 ...
So the kernel also maintains a **hash** table of all memory-resident pages. ... Interaction of **cache** and virtual memory Our pipeline drawings for the "Y86" ...
www.cs.rochester.edu/u/scott/252/notes/10_vm - 41k - Cached - Similar pages

[PDF] Microsoft PowerPoint - Lecture21.ppt

File Format: PDF/Adobe Acrobat - View as HTML
as many **hash** table entries as. there are pages in memory. Virtual Address. Page. Offset.
Hash ... One solution – always **translate** before going to the **cache ...**
www-csl.csres.utexas.edu/users/billmark/teach/cs352-05-spring/lectures/Lecture21.pdf -
Similar pages

Result Page: **Previous** 1 **2** 3 4 5 6 7 8 9 10 11 **Next**

translation cache hash Search

Search within results | Language Tools | Search Tips

Google Home - Advertising Programs - Business Solutions - About Google